



Правительство Санкт-Петербурга
Комитет по образованию

Государственное бюджетное общеобразовательное учреждение
Лицей №126
Калининского района Санкт-Петербурга

Принята

на педагогическом совете

Протокол № 1 от 30 августа 2024 г.

Утверждаю

Директор _____ А. А. Рагимова

Приказ № 351 от 30 августа 2024 г.

Дополнительная общеразвивающая программа

«Программирование на языке "Python":
"Основы промышленного программирования"»

возраст обучающихся 14 - 18 лет
срок освоения: 36 недель

Составители программы:
Крылатых Данил Николаевич,
педагог дополнительного образования

Санкт-Петербург

2024

I. ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Основные характеристики программы

Дополнительная общеразвивающая программа «Программирование на языке "Python": "Основы промышленного программирования"» (далее - ДООП) имеет **техническую направленность**.

Адресат программы

Дополнительная общеразвивающая программа «Программирование на языке "Python": "Основы промышленного программирования"» предполагает начальный уровень компьютерной грамотности, к занятиям в группах допускаются лица в возрасте 14 - 18 лет.

Актуальность ДОП

Актуальность программы обусловлена высоким интересом подростков к IT-сфере. Важнейшей характеристикой подрастающего поколения является активность в информационном пространстве, интернет-коммуникации. Python является очень востребованным языком программирования - он отлично подходит для знакомства с различными современными парадигмами программирования и активно применяется в самых разных областях от разработки веб-приложений до машинного обучения. Программа «Программирование на языке "Python": "Основы промышленного программирования"» в связи с востребованностью на рынке, является своего рода уникальным образовательным продуктом в области информационных технологий. В условиях динамично развивающейся сферы информационных технологий, потребности кадров, развития инфраструктуры программа соответствует региональным социально-экономическим и социокультурным потребностям и проблемам региона.

Программа разработана в соответствии со следующими документами:

- Федеральный закон от 29 декабря 2012 г. № 273-ФЗ «Об образовании в Российской Федерации», (далее - ФЗ № 273);
- Концепция развития дополнительного образования детей до 2030 года (распоряжение Правительства РФ от 31 марта 2022 г. N 678-р), (далее – Концепция);
- Постановление Главного государственного санитарного врача от 28.09.2020 г. № 28 «Об утверждении санитарных правил СП 2.4.3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи»;
- Постановление Главного государственного санитарного врача Российской Федерации от 28.01.2021 № 2 «Об утверждении санитарных правил и норм СанПиН 1.2.3685- 21 «Гигиенические нормативы и требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания» (рзд.VI. Гигиенические нормативы по устройству, содержанию и режиму работы организаций воспитания и обучения, отдыха и оздоровления детей и молодежи»)
- Устав Государственного бюджетного общеобразовательного учреждения Лицей № 126 Калининского района Санкт-Петербурга и другими действующими нормативно-правовыми актами с учетом возрастных и индивидуальных особенностей обучающихся на занятиях.

ДОП ежегодно обновляется с учетом развития науки, техники, культуры, экономики, технологий и социальной сферы.

Отличительная особенность ДОП

Отличительная особенность заключается в том, что программа составлена таким образом, чтобы обучающиеся могли овладеть всем комплексом знаний по организации исследовательской деятельности, выполнению проектной работы, познакомиться с требованиями, предъявляемыми к оформлению и публичному представлению результатов

своего труда, а также приобрести практические навыки работы по созданию социально значимых проектов виде программы, написанной на языке программирования Python.

Уровень освоения ДОП

Дополнительная общеразвивающая программа «Программирование на языке "Python": "Основы промышленного программирования"» рассчитана на общекультурный уровень освоения программы.

Объем и срок освоение программы

Общее количество учебных часов, запланированных на весь период обучения: 144 академических часа. Срок реализации программы 36 недель.

Цель и задачи ДОП

Цель программы: подготовка к использованию как языка программирования, так и методов программирования на Python в учебной и последующей профессиональной деятельности в различных предметных областях.

ДООП «Программирование на языке "Python": "Основы промышленного программирования"» ставит перед собой следующие **задачи**:

Обучающие (предметные):

- познакомить с конструкциями языка программирования Python;
- познакомить с принципами и методами функционального и объектно-ориентированного программирования;
- познакомить с основными структурами данных и типовыми методами обработки этих структур;
- сформировать навыки работы в интегрированной среде разработки на языке Python;
- сформировать навыки разработки эффективных алгоритмов и программ на основе изучения языка программирования Python;
- сформировать и развить навыки алгоритмического и логического мышления, грамотной разработки программ;

Развивающие (метапредметные):

- способствовать развитию навыков проектной деятельности;
- способствовать развитию навыков поиска информации в сети Интернет, анализа выбранной информации на соответствие запросу, использования информации при решении задач;
- сформировать у обучающихся интерес к программированию, самостоятельность и творческий подход к решению задач с использованием средств вычислительной техники;
- способствовать развитию навыков работы с различными источниками информации, необходимой для решения учебных задач;
- познакомить с правилами индивидуального и коллективного безопасного поведения при работе с компьютерной техникой;
- способствовать развитию умения планировать свои действия с учётом фактора времени, в обстановке с элементами конкуренции, предвидеть результат и достигать его, при необходимости вносить коррективы в первоначальный замысел.

Воспитательные (личностные):

- способствовать воспитанию этики групповой работы;
- сформировать коммуникативную культуру обучающихся, как внутри проектных групп, так и в коллективе в целом;
- способствовать воспитанию упорства в достижении результата;
- сформировать целеустремлённость, организованность, неравнодушие,

ответственное отношение к труду и уважительное отношение к окружающим.

Планируемые результаты освоения программы

Личностные:

- развита способность к саморазвитию и самообразованию, любознательность, внимательность и настойчивость при выполнении заданий практического характера;
- сформированы навыки организации учебного сотрудничества и совместной деятельности с учителем и сверстниками в процессе проектной и учебно-исследовательской деятельности;
- сформировано ответственное отношение к учению, готовность и способность обучающихся к саморазвитию и самообразованию, средствами информационных технологий;
- развита коммуникативная компетентность в общении и сотрудничестве со сверстниками в процессе образовательной, учебно-исследовательской и проектной деятельности;
- развита база целостного мировоззрения, соответствующего современному уровню развития информационных технологий.

Метапредметные:

- сформированы навыки инженерного мышления, умения работать по предложенным инструкциям, программирования, проектирования и эффективного использования электронного вычислительного оборудования;
- развиты внимательность, аккуратность и изобретательность при работе с техническими устройствами, разработке приложений и выполнении учебных проектов;
- сформированы творческое мышление и воображение, умение излагать мысли в четкой логической последовательности, отстаивать свою точку зрения, грамотно работать с критикой и извлекать из неё пользу, анализировать ситуацию и самостоятельно находить ответы на вопросы путем логических рассуждений и информационного поиска.

Предметные:

- получены знания основ современных языков программирования;
- сформированы умения объяснять и использовать на практике как простые, так и сложные структуры данных и конструкции для работы с ними;
- сформированы умения искать и обрабатывать ошибки в коде;
- сформированы умения разбивать решение задачи на подзадачи;
- развиты способности написания грамотного, красивого кода;
- развиты способности анализа, как своего, так и чужого код;
- развиты способности работы с информацией: находить, оценивать и использовать информацию из различных источников, необходимую для решения профессиональных задач (в том числе на основе системного подхода);
- развиты способности самостоятельного изучения новых технологий;
- развиты способности грамотной коммуникации, исходя из целей и ситуации;
- развиты способности работы в команде и использования соответствующих технологий для организации командной работы;
- получен опыт участия в проектных конкурсах по программированию, как единолично, так и в составе команды

Теоретическая подготовка включает: владение теоретическими знаниями (по основным разделам учебного плана программы), системой понятий; владение специальной терминологией; знание правил и алгоритмов деятельности.

Практическая подготовка ребенка включает: практические умения и навыки, предусмотренные программой; владение специальным оборудованием и оснащением; творческие навыки.

Организационно-педагогические условия реализации программы

Язык реализации программы – образовательная деятельность по дополнительной общеразвивающей программе «Программирование на языке "Python": "Основы промышленного программирования"» осуществляется на русском языке.

Форма обучения: очная.

Особенности реализации программы: реализация программы проходит в течение всего учебного года, включая каникулярное время, с использованием электронного обучения и дистанционных образовательных технологий.

Особенности организации образовательного процесса: образовательный процесс по реализации ДОП «Программирование на языке "Python": "Основы промышленного программирования"» строится на принципах научности, доступности, наглядности, систематичности и последовательности, связи теории с практикой, сознательности и активности обучения, индивидуальном подходе в обучении с учётом психолого-педагогических особенностей каждого обучающегося.

Условия набора в коллектив: набор в объединение осуществляется по желанию при отсутствии медицинских противопоказаний.

Условия формирования групп: группы формируются разновозрастные.

Количество детей в группе: 12 человек.

Режим занятий: 2 раза в неделю по 2 часа – 144 академических часа в год.

Формы организации занятий: занятия проводятся по группам, программой предусматриваются как аудиторные, так и внеаудиторные формы организации занятий.

Формы проведения занятий: основной формой организации деятельности является учебное занятие традиционной формы.

Формы организации деятельности учащихся на занятии: фронтальная, индивидуальная, групповая.

Материально-техническое оснащение ДОП:

- проектор и экран для демонстрации учебного материала;
- персональные компьютеры для обучающихся с доступом к сети Интернет;
- операционная система Windows
- среда программирования Python

II. УЧЕБНЫЙ ПЛАН

№ п/п	Название раздела, темы	Кол-во часов			Формы контроля /аттестации
		всего	теория	практика	
1.	Повторение основ программирования.	8	4	4	Устный опрос, беседа, решение задач
2.	Введение в промышленное программирование	136	46	90	Контрольные задания Учебное тестирование
ИТОГО		144	50	94	

Утверждаю

Директор _____ А. А. Рагимова

Приказ № 351 от 30 августа 2024 г.

III. КАЛЕНДАРНЫЙ УЧЕБНЫЙ ГРАФИК

Календарный учебный график реализации дополнительной
общеразвивающей программы «Программирование на языке "Python": "Основы
промышленного программирования"»
на 2024 - 2025 учебный год

Педагог: Крылатых Данил Николаевич

Год обучения	Дата начала занятий	Дата окончания занятий	Количество учебных недель	Количество во учебных дней	Количество во учебных часов	Режим занятий
1 год	01.09.24	31.05.25	36	72	144	2 раз в неделю по 2 часа

III. РАБОЧАЯ ПРОГРАММА

Содержание программы

ТЕМА 1. Повторение основ программирования.

Теория: Инструктажи по технике безопасности. Повторение основных конструкций языка. Повторение чтения и вывода (input, print), алгоритмических конструкций (if, elif, for, while), типов данных (int, str, float), контейнеров (list, set, dict), а также операций над ними, функций и методов. Создание своих классов, атрибуты (в том числе методы) класса, «магические методы». Наследование, перегрузка методов.

Практика: Решение задач по теме «Повторение». Решение задач на основные конструкции и структуры данных. Решение задач на классы и библиотеки.

ТЕМА 2. Введение в промышленное программирование

Теория: PyQT. QTDesigner. Основные понятия. Файлы в Python. Типы файлов и работа с ними. Диалоги, работа с изображениями. Работа с простыми таблицами (csv). Работа с SQL-таблицами и отображение данных в PyQT. Введение в репозитории. Работа с удаленными репозиториями. Создание игр в PyGame. WEB. Работа с популярными форматами файлов. Работа с протоколом HTTP. Введение во flask. Чат-боты.

Практика: Работа над проектом по теме QT. Работа над проектом по теме PyGame. Разработка итогового проекта. Решение задач.

V. МЕТОДИЧЕСКИЕ и ОЦЕНОЧНЫЕ МАТЕРИАЛЫ

5.1. Методические материалы

УМК программы состоит из трех компонентов:

1. Учебные и методические пособия для педагога и учащихся.
2. Система средств обучения.
3. Система средств контроля результативности обучения.

Учебные и методические пособия для педагога и учащихся

1. Python для детей. Самоучитель по программированию / Джейсон Бриггс ; пер. с англ. Станислава Ломакина ; [науч. ред. Д. Абрамова]. — М. : Манн, Иванов и Фербер, 2017 — 320 с.; 2.
2. К. Вордерман и др. Программирование для детей: Иллюстрированное руководство по языкам Scratch и Python, 224 стр. Издательство: Манн, Иванов и Фербер, 2017 г.; 3.
3. Б. Пэйн. Python для детей и родителей, 352 стр. Издательство: Эксмо, 2017
4. Материалы и презентации к урокам в LMS Яндекс. Лицея.

5.2. Оценочные материалы

Для отслеживания результативности образовательной деятельности по программе проводятся: входной, текущий, промежуточный и итоговый.

Входной контроль – оценка стартового уровня образовательных возможностей учащихся при поступлении в объединение, ранее не занимающихся по данной дополнительной общеобразовательной общеразвивающей программе. Входная диагностика проводится в сентябре с целью выявления первоначального уровня знаний и умений.

Входной контроль проводится в форме тестирования.

Текущий контроль – оценка уровня и качества освоения тем программы и личностных качеств учащихся осуществляется на занятиях в течение всего учебного года для отслеживания уровня освоения учебного материала программы и развития ИТ-компетенций. Текущий контроль осуществляется посредством наблюдения и беседы.

Промежуточный контроль – оценка уровня и качества освоения учащимися дополнительной общеобразовательной общеразвивающей программы по итогам изучения раздела.

Итоговый контроль – оценка и качество освоения учащимися дополнительной общеобразовательной общеразвивающей программы по завершению учебного года. Итоговый контроль проходит в форме защиты итоговых проектных работ.

Формы фиксации результатов:

- диагностическая карта определения результатов по программе «Основы промышленного программирования» (Приложение №1);
- анкета для родителей «Отношение родительской общественности к качеству образовательных услуг и степень удовлетворенности образовательным процессом в объединении»;
- фотографии защиты проектов.

Утверждаю
Директор _____ А. А. Рагимова

Приказ № 351 от 30 августа 2024 г.

Календарно-тематический план

№ п/п	Тема занятия	Кол-во часов	Дата занятия	
			план	факт
1.	Вводное занятие. Введение в программу 2-го года обучения Инструктаж по ТБ	2	1 неделя	
2.	Повторение. Решение задач на основные конструкции и структуры данных.	2	1 неделя	
3.	Повторение. Решение задач на классы и библиотеки	2	2 неделя	
4.	Решение задач по теме "Повторение"	2	2 неделя	
5.	QT 1. Что такое PyQt	2	3 неделя	
6.	QT 2. QtDesigner. Основные понятия	2	3 неделя	
7.	QT 2. QtDesigner. Решение задач	2	4 неделя	
8.	QT 3. Файлы в Python. Типы файлов и работа с ними	2	4 неделя	
9.	QT 3. Внутреннее устройство файлов	2	5 неделя	
10.	QT 4. Диалоги, работа с изображениями	2	5 неделя	
11.	QT 4. Диалоги, работа с изображениями	2	6 неделя	
12.	QT 5. Работа с простыми таблицами (csv)	2	6 неделя	
13.	QT 5. Работа с табличными данными в PyQt	2	7 неделя	
14.	QT 6. Введение в БД. Часть 1	2	7 неделя	
15.	QT 6. Введение в БД. Часть 2	2	8 неделя	
16.	QT 7. Работа с SQL-таблицами и отображение данных в PyQt. Часть 1	2	8 неделя	
17.	QT 7. Работа с SQL-таблицами и отображение данных в PyQt. Часть 2	2	9 неделя	
18.	QT 8. Обработка событий	2	9 неделя	
19.	QT 8. Сборка независимого приложения	2	10 неделя	
20.	Работа над проектом по теме QT	2	10 неделя	
21.	Защита проекта по теме QT	2	11 неделя	
22.	Введение в репозитории. Подключение PyCharm	2	11 неделя	
23.	Введение в репозитории. Работа с удаленными репозиториями	2	12 неделя	
24.	PyGame 1. Введение	2	12 неделя	
25.	PyGame 1. Примитивные объекты	2	13 неделя	
26.	PyGame 2. Игровой цикл	2	13 неделя	
27.	PyGame 2. События	2	14 неделя	
28.	PyGame 3. Основные команды при одиночной работе с Git	2	14 неделя	
29.	PyGame 3. Основные команды при одиночной работе с Git	2	15 неделя	
30.	PyGame 4. Клеточное поле. Отображение	2	15 неделя	
31.	PyGame 4. Клеточное поле. Преобразование	2	16 неделя	
32.	PyGame 5. Классические игры на клеточном поле	2	16 неделя	

33.	PyGame 5. Классические игры на клеточном поле	2	17 неделя	
34.	PyGame 6. Изображения	2	17 неделя	
35.	PyGame 6. Спрайты	2	18 неделя	
36.	PyGame 7. Столкновения	2	18 неделя	
37.	PyGame 7. Взаимодействия объектов игры	2	19 неделя	
38.	PyGame 8. Игра в целом	2	19 неделя	
39.	PyGame 8. Игра в целом	2	20 неделя	
40.	PyGame 9. Украшение игры	2	20 неделя	
41.	PyGame 9. Украшение игры	2	21 неделя	
42.	Цели и подходы к тестированию. Создание "самодельных" тестов (без библиотек)	2	21 неделя	
43.	Работа над проектом по теме PyGame	2	22 неделя	
44.	Защита проекта по теме PyGame	2	22 неделя	
45.	WEB. Работа с популярными форматами файлов (json, xml)	2	23 неделя	
46.	WEB. Работа с популярными форматами файлов (json, xml)	2	23 неделя	
47.	WEB. Знакомство с API	2	24 неделя	
48.	WEB. Знакомство с API	2	24 неделя	
49.	WEB. Понятие исключения, обработка исключений	2	25 неделя	
50.	WEB. Собственные исключения	2	25 неделя	
51.	WEB. Работа с командной строкой (скрипты, аргументы)	2	26 неделя	
52.	WEB. Периодические задачи (модуль shedule)	2	26 неделя	
53.	WEB. Библиотека argparse.	2	27 неделя	
54.	WEB. Задачи на создание скриптов с помощью argparse	2	27 неделя	
55.	WEB. Работа с протоколом HTTP	2	28 неделя	
56.	WEB. Работа с протоколом HTTP	2	28 неделя	
57.	WEB. Решение задач на API Яндекс-карт	2	29 неделя	
58.	WEB. Решение задач на API Яндекс-карт	2	29 неделя	
59.	WEB. Введение во flask	2	30 неделя	
60.	WEB. Обработка html-форм	2	30 неделя	
61.	WEB. Шаблоны	2	31 неделя	
62.	WEB. Flask-WTF	2	31 неделя	
63.	WEB. Знакомство с flask- sqlalchemy	2	32 неделя	
64.	WEB. Знакомство с flask- sqlalchemy	2	32 неделя	
65.	WEB. REST-API. Основные понятия	2	33 неделя	
66.	WEB. REST-API. Flask-restfull	2	33 неделя	
67.	WEB. Разворачиваем проект в облаке	2	34 неделя	
68.	WEB. Алиса	2	34 неделя	
69.	Чат-боты. Вконтакте	2	35 неделя	
70.	Чат-боты. Телеграмм	2	35 неделя	
71.	Разработка итогового проекта	2	36 неделя	
72.	Защита итогового проекта	2	36 неделя	

Диагностическая карта определения результатов по программе

за учебный год _____

Группа:

Педагог:

Даты проведения: вход:

1 полугодие:

2 полугодие _____

№	Фамилия	Входной контроль	Промежуточный контроль	Итоговый контроль
1.				
2.				
3.				
4.				
5.				
6.				
7.				
8.				
9.				
10.				
11.				
12.				

Уровень освоения обучающимися программы

Параметры результативности	Входной контроль (количество / %)	Промежуточный контроль (количество / %)	Итоговый контроль (количество / %)
Высокий уровень			
Средний уровень			
Начальный уровень			

Входной контроль

по программе дополнительного образования

«Программирование на языке "Python": "Основы промышленного программирования"»

ФИО _____

Группа _____

Решите 4 задачи:

1. Вторая сенсация.

Агентства новостей такого не любят. Все хорошо в меру. Всамделишный звездолет, приземлившийся среди бела дня в центре Лондона, не мог не стать сенсацией высшего разряда. Второй, совершенно на него не похожий, приземлившийся тремя с половиной часами позже, сенсацией уже не стал.

Напишите программу для поиска НЕ сенсаций.

В каждой строке найдите слово, стоящее после самого длинного (и самого первого самого длинного, если их несколько) в этой строке. Соберите все такие слова без повторов и выведите по одному в строке в любом порядке. Гарантируется, что самое длинное – не самое последнее.

2. Вероятные события.

Далеко-далеко, в чернильных глубинах космоса, вдали от чьих-либо глаз, происходили некие события.

Незримые обитателям таинственно, непредсказуемо капризного сектора «Множественное Зет» (того самого, в сердцевине которого находится бесконечное множество вероятностных вариантов планеты под названием «Земля»), они тем не менее имели для этих обитателей самые серьезные последствия.

Напишите функцию, собирающую близкие по вероятности копии в одну группу.

Функция `events()` принимает произвольное число аргументов-строк и именованный аргумент `chance` со значением по умолчанию `Earth`.

Функция должна вернуть словарь, каждый ключ которого – количество общих букв строки со значением аргумента `chance`, причем одинаковые буквы считаются столько раз, сколько их есть в строке, а вот регистр не важен. Значение по ключу – список строк с таким количеством общих букв, записанных без повторов в обратном алфавитном порядке.

3. В ответе за всё.

Напишите функцию для фотофиксации.

Функция `response()` принимает аргументы: имя файла изображения, 4 кортежа по два числа и именованный аргумент `delta` со значением по умолчанию 50.

Функция сначала подготавливает исходное изображение: отражает относительно горизонтальной оси и поворачивает на 90 градусов против часовой стрелки; затем все пиксели, у которых разница между красной и синей компонентами превышает значение `delta`, выравнивает – половину `delta` отнимает у большего значения, и половину `delta` добавляет меньшему.

Затем из изображения вырезаются 4 прямоугольника 600x150 по координатам левых верхних углов, записанных в кортежах – аргументах функции. Эти прямоугольники вставляются один под другим на новое изображение, которое функция и возвращает.

4. Заключительный акт.

Напишите класс Calculator, моделирующий работу суперкомпьютера.

Экземпляр класса при инициализации принимает произвольное число задач для выполнения и именованный аргумент form (класс вычислителя, принимает значения 0 или 1) со значением по умолчанию 0.

Класс обеспечивает функциональность:

- экземпляр класса можно разделить на число: $arg = cl / n$, возвращается список новых экземпляров класса, между которыми задачи для вычисления распределены поровну – по целой части от деления длины перечня на число (кроме, возможно, последнего – к нему добавляются все оставшиеся задачи), а классы чередуются – сначала такой же, как у исходного экземпляра, затем другое возможное значение, и так далее); исходный экземпляр не меняется;
- к экземпляру можно прибавить число: $cl += n$, класс экземпляра меняется на остаток от деления на 2 суммы исходного класса и числа n ;
- экземпляр можно вызвать с аргументом-строкой: эта строка добавляется в начало перечня задач;
- для печати возвращается строка вида:
- `Calculator(form: <класс>, results: <результаты вычислений выражений из перечня задач через запятую и пробел>)`

Промежуточный контроль
по программе дополнительного образования
«Программирование на языке "Python": "Основы промышленного программирования"»

Задание: создайте в группе игру с использованием PyGame в соответствии с приведёнными ниже критериями (см. таблицу 1).

Таблица 1.

№	Критерии	Баллы
1.	Объём кода (500 и выше)	2 балла
2.	Чистота кода	2 балла
3.	Качество проектирования	2 балла
4.	Использование User Interface элементы (UI), которые должны сообщать о ключевых событиях (как минимум, старт и завершение сценария)	2 балла
5.	Использование спрайтов и анимации	2 балла
6.	Реализовано несколько уровней	2 балла
7.	Работоспособность	2 балла
8.	Оригинальность идеи	2 балла
		16 баллов

Критерии оценивания
промежуточного контроля
по программе дополнительного образования
«Программирование на языке "Python": "Основы промышленного программирования"»

По итогам выполнения группового проекта возможно набрать 16 баллов.

Высокий уровень освоения программы курса - 12 – 16 баллов.

Средний уровень освоения программы курса - 6 – 11 баллов.

Низкий уровень освоения программы курса – 5 и менее баллов.

Итоговый контроль
по программе дополнительного образования
«Программирование на языке "Python": "Основы промышленного программирования"»

Задание: разработайте приложение с использованием технологий изученных в данном курсе:

Таблица 1.

№	Критерии	Мак балл	Оценка
1.	Объём программного кода	15	
2.	Чистота кода (Оценивается PEP 8, константы в коде, сору-paste, говорящие имена переменных и так далее.)	5	
3.	Качество проектирования (Используются ли классы, модули и прочее. Если код написан очень примитивно, то оценка снижается.)	20	
4.	Применённые технологии (Реализованы ли практически все изученные в модуле технологии.)	20	
5.	Оригинальная идея	10	
6.	Соблюдение сроков	15	
7.	Работоспособность	1	
8.	Защита	10	
9.	Дополнительный балл	4	
		100 баллов	

Критерии оценивания
итогового контроля
по программе дополнительного образования
«Программирование на языке "Python": "Основы промышленного программирования"»

Высокий уровень освоения программы курса - 85 – 100 баллов.

Средний уровень освоения программы курса - 65 – 84 баллов.

Низкий уровень освоения программы курса – 64 и менее баллов.

Приложение 3.**Перечень интеллектуальных и творческих конкурсов, мероприятий, направленных на развитие интеллектуальных и творческих способностей, интереса к научной (научно-исследовательской), инженерно-технической, изобретательской деятельности**

№ п/п	Название мероприятия	Уровень мероприятия	Возрастная категория	Примерные сроки проведения
1.	НТО	Всероссийский	12 – 18	Сентябрь – ноябрь
2.	«Pythonic Mastery»	Городской	13 – 18	Декабрь
3.	«Codeforces»	Всероссийский	12 – 18	Январь – март
4.	Открытый международный фестиваль по робототехнике и программированию «Red fest»	Региональный	10 – 18	Февраль – апрель

